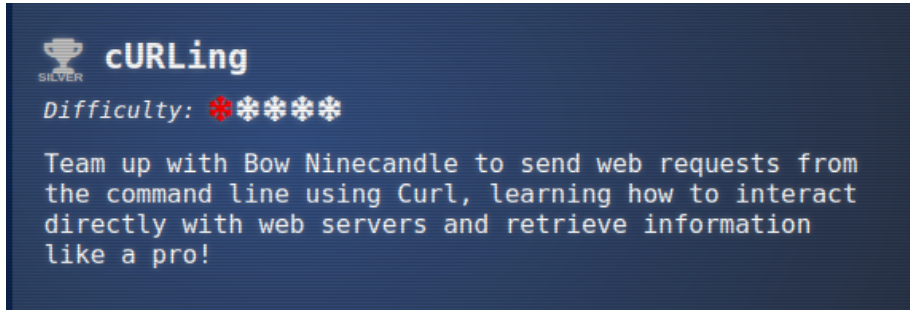


Objective “cURLing”

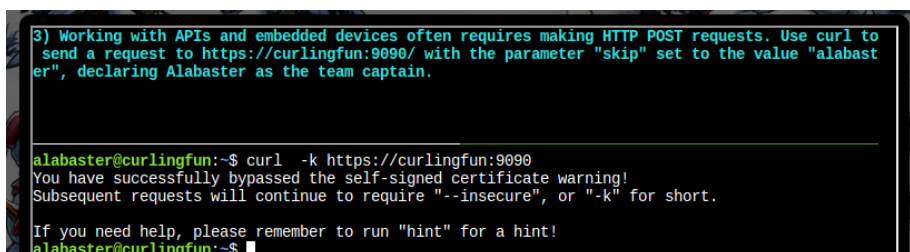
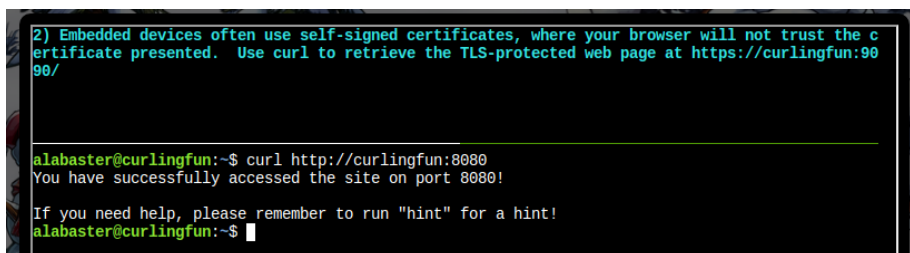
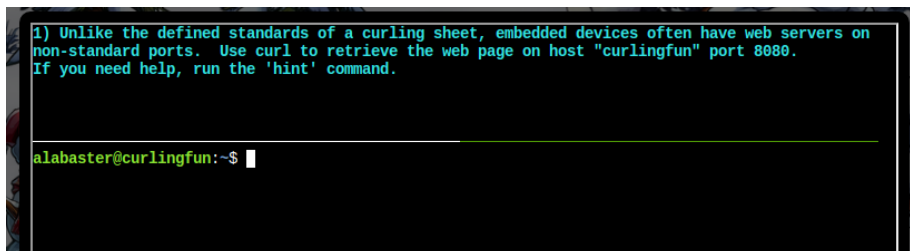


Location

North Pole Monitoring Station

Task and Solution

The task is getting to know the CLI tool “curl” and its features



4) Working with APIs and embedded devices often requires maintaining session state by passing a cookie. Use curl to send a request to https://curlingfun:9090/ with a cookie called "end" with the value "3", indicating we're on the third end of the curling match.

```
alabaster@curlingfun:~$ curl -k -X POST -d "skip=alabaster" "https://curlingfun:9090/"
You have successfully made a POST request!
alabaster@curlingfun:~$
```

5) Working with APIs and embedded devices sometimes requires working with raw HTTP headers. Use curl to view the HTTP headers returned by a request to https://curlingfun:9090/

```
alabaster@curlingfun:~$ curl -k -X POST --cookie "end=3" "https://curlingfun:9090/"
You have successfully set a cookie!
alabaster@curlingfun:~$
```

6) Working with APIs and embedded devices sometimes requires working with custom HTTP headers. Use curl to send a request to https://curlingfun:9090/ with an HTTP header called "Stone" and the value "Granite".

```
* TLSv1.2 (OUT), TLS header, Supplemental data (23):
* TLSv1.3 (OUT), TLS handshake, Finished (20):
* SSL connection using TLSv1.3 / TLS_AES_256_GCM_SHA384
* ALPN, server accepted to use http/1.1
* Server certificate:
*  subject: C=US; ST=Some-State; O=Internet Widgits Pty Ltd; CN=localhost
*   start date: Feb  7 16:23:39 2024 GMT
*   expire date: Feb  6 16:23:39 2025 GMT
*   issuer: C=US; ST=Some-State; O=Internet Widgits Pty Ltd; CN=localhost
*   SSL certificate verify result: self-signed certificate (18), continuing anyway.
* TLSv1.2 (OUT), TLS header, Supplemental data (23):
> GET / HTTP/1.1
> Host: curlingfun:9090
> User-Agent: curl/7.81.0
> Accept: */*
>
* TLSv1.2 (IN), TLS header, Supplemental data (23):
* TLSv1.3 (IN), TLS handshake, Newsession Ticket (4):
* TLSv1.2 (IN), TLS header, Supplemental data (23):
* TLSv1.3 (IN), TLS handshake, Newsession Ticket (4):
* old SSL session ID is stale, removing
* TLSv1.2 (IN), TLS header, Supplemental data (23):
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Server: nginx/1.18.0 (Ubuntu)
< Date: Fri, 03 Jan 2025 16:20:58 GMT
< Content-Type: text/plain; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< Custom-Header: You have found the custom header!
<
You have successfully bypassed the self-signed certificate warning!
Subsequent requests will continue to require "--insecure", or "-k" for short.

If you need help, please remember to run "hint" for a hint!
* Connection #0 to host curlingfun left intact
alabaster@curlingfun:~$ curl -k -v "https://curlingfun:9090/"
```

```
[curlingfun]: <XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX>
```

7) curl will modify your URL unless you tell it not to. For example, use curl to retrieve the following URL containing special characters: `https://curlingfun:9090/../../../../etc/hacks`

```
alabaster@curlingfun:~$ curl -k -H "Stone: Granite" "https://curlingfun:9090/"
You have successfully set a custom HTTP header!
alabaster@curlingfun:~$
```

Great work!

Once HHC grants your achievement, you may close this terminal.

```
alabaster@curlingfun:~$ curl -k --path-as-is "https://curlingfun:9090/../../../../etc/hacks"
You have successfully utilized --path-as-is to send a raw path!
alabaster@curlingfun:~$
```